**LAPORAN WEBINAR**

**Mobile programing**

**(TIPE Data dan Collection pada kotlin)**

****

**Disusun Oleh:**

**LIDIA FITA SARI**

**3042019052**

**TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI KETAPANG**

**TAHUN AKADEMIK**

**2020/2021**

**KOTLIN**

Kotlin adalah sebuah bahasa pemrograman dengan [pengetikan statis](https://id.wikipedia.org/w/index.php?title=Pengetikan_statis&action=edit&redlink=1" \o "Pengetikan statis (halaman belum tersedia)) yang berjalan pada [Mesin Virtual Java](https://id.wikipedia.org/wiki/Java_Virtual_Machine" \o "Java Virtual Machine) ataupun menggunakan kompiler [LLVM](https://id.wikipedia.org/w/index.php?title=LLVM&action=edit&redlink=1) yang dapat pula dikompilasikan kedalam bentuk [kode sumber](https://id.wikipedia.org/wiki/Kode_sumber" \o "Kode sumber) JavaScript. Pengembang utamanya berasal dari tim programer dari JetBrains yang bermarkas di Rusia. Meskipun sintaksisnya tidak kompatibel dengan bahasa [Java](https://id.wikipedia.org/wiki/Java), Kotlin didesain untuk dapat bekerja sama dengan kode bahasa Java dan bergantung kepada kode bahasa Java dari [Kelas Pustaka Java](https://id.wikipedia.org/w/index.php?title=Java_Class_Library&action=edit&redlink=1" \o "Java Class Library (halaman belum tersedia)) yang ada, seperti berbagai framework Java yang ada. Tim Pengembang memutuskan menamakannya Kotlin dengan mengambil nama dari [sebuah pulau di Rusia](https://id.wikipedia.org/wiki/Pulau_Kotlin" \o "Pulau Kotlin), sebagaimana Java yang mengambil nama dari pulau Jawa di Indonesia. Setelah Google mengumumkan bahwa Kotlin menjadi bahasa kelas satu bagi Android, maka bersama Java dan C++, Kotlin menjadi bahasa resmi untuk pengembangan aplikasi-aplikasi Android.

Kotlin versi 1.0 dirilis pada 15 Februari 2016 [[5]](https://id.wikipedia.org/wiki/Kotlin_(bahasa_pemrograman)#cite_note-5) Versi ini secara resmi ditetapkan sebagai versi rilis stabil pertama dan JetBrains telah menetapkan dukungan versi sebelumnya untuk jangka panjangan dengan versi ini. Pada [Google I/O](https://id.wikipedia.org/wiki/Google_I/O) 2017, [Google](https://id.wikipedia.org/wiki/Google) mengumumkan dukungan kelas pertama untuk Kotlin pada [Android](https://id.wikipedia.org/wiki/Android_(sistem_operasi)). Kotlin versi 1.2 dirilis pada 28 November 2017.[[6]](https://id.wikipedia.org/wiki/Kotlin_(bahasa_pemrograman)" \l "cite_note-6) Fitur berbagi kode antara JVM dan platform Javascript baru ditambahkan pada versi rilis ini. Kotlin versi 1.3 dirilis pada 29 Oktober 2018, membawa coroutines pada pemrograman asynchronous.

1. **Tipe data Kotlin**

Kotlin adalah bahasa pemrograman modern yang mengkompilasi Java ke bytecode. Bahasa yang bebas dan [open source](https://github.com/jetbrains/kotlin), dan menjanjikan untuk membuat coding untuk Android bahkan lebih menyenangkan.Ada dua macam tipe data, yang pertama adalah tipe data primitif (int, boolean, long, char, dll) dan kedua adalah adalah tipe data reference (array, String, dll). Tipe data primitif adalah tipe data yang hanya menyimpan satu nilai pada setiap satu variabel. Sebaliknya tipe data reference dapat menyimpan banyak nilai. Kotlin membuat semua tipe data tersebut bertindak sebagai objek. Berikut beberapa basis tipe data yang diketahui:

|  |  |  |
| --- | --- | --- |
| Char | Hanya satu karakter | “A” |
| String | Kumpulan dari banyak karakter (teks) | “kotlin” |
| Int, Long, Short, Byte | Angka | 7 |
| Float, Double | Angka pecahan | 7.6 |
| Boolean | Hanya memiliki 2 nilai | True/false |
| Array | Menyimpan banyak objek | arrayOf(“aku”, ’kamu”,”dia”) |

Table 1. tipe data

1. **Cara menulis variable**

Tipe data dan variabel merupakan kedua hal yang tidak bisa dilepas ketika kita belajar bahasa pemrograman apapun. Akan tetapi, tidak semua tipe data dan variabel bisa dikatakan sama. Karena setiap bahasa pemrograman memiliki aturan yang berbeda-beda. Termasuk bahasa Kotlin ini, hal yang pertama kita akan pelajari adalah tipe data dan variabel. Kita akan mengulang pengertian tipe data dan variabel. Variabel adalah simbol yang digunakan untuk menyimpan sebuah nilai. Sedangkan tipe data adalah jenis nilai yang akan disimpan.

**var/val nama Variabel : TipeData = isi(value)**

contoh :

**var company: String = “dicoding”**

atau bias disingkat

**var company = “dicoding”**

1. **perbedaan var dengan val**

Dalam bahasa pemrograman penggunaan variabel biasa dilakukan untuk menyimpan suatu informasi atau nilai didalam program kita. Pada bahasa kotlin terdapat 2 kata kunci yang dapat digunakan saat mendeklarasikan sebuah variabel yaitu kata kunci var dan val.

|  |  |
| --- | --- |
| Var vs Val | |
| Bisa diubah nilainya (mutable) | Tidak bias diubah nilainya (immutable) |
| var company = “dicoding” | val company = “dicoding” |
| Company = ”dcoding academy” | Company = “decoding academy” // val cannot be reassigned |

Table 2. var vs val

1. **nullable types**

Null adalah type yang digunakan untuk mewakili kondisi yang tidak mempunyai nilai.mMisal adalah kita bekerja pada data pemain, misal pemain mempunyai field nama, umur dan klub. Nama dan umur keduanya harus mempunyai nilai karena semua pemain mempunyainya, namun tidak semua pemain punya klub maka kita perlu suatu nilai untuk mewakili kondisi ini.mAda 2 type pada kotlin, non-null type dan nullable type.

* non-null type tidak bisa menyimpan null, jadi type ini pasti mempunyai nilai.
* nullable type bisa menyimpan nilai maupun null.

Nullable type bisa dianalogikan sebagai suatu kotak, kotak ini bisa berisi nilai ataupun tidak. Ketika kotak itu tidak berisi nilai, kotak itu berisi null. Kotaknya sendiri selalu ada.

Contoh :

Val text : string? = null

Contoh salah :

Val text : string = null

// compile time error

Ada 2 pilihan untuk menggunakan nullable type yaitu dengan Safe Call Operator?. atau menggunakan Elvis Operator?:

1. Safe Call Operator?.

Memanggil nullable dengan aman

Val text : string?=null

Val textLength=text?.length

1. Elvis Operator

Nilai default jika objek bernilai null

Val text:string?=null

Val textlength=text?.length?:0

1. **Collection**

Collection merupakan kumpulan dari banyak data. Collection terbagi menjadi 3 diantaranya yaitu :

1. List

List adalah salah satu collection yang bisa digunakan untuk meniman nilai bebas dengan index (boleh sama)

Val integerList = listOF(4, 2, 1, 5, 1, 2,) // Output : [4, 2, 1, 5, 1, 2]

**Bisa interritasList[3]**

1. Set

Set adalah collection yang digunakan untuk menyimpan nilai yang unik.

Val integerSet = setoff(4, 2, 1, 5, 1, 2) // Output : [4, 2, 1, 5]

**Tidak bisa integerset[3]**

1. Maps

Digunakan untuk menyimpan dengan format key to value, key unik tidak boleh sama.

Val capital = mapOF(

“Jakarta” to “Indonesia”,

“London” to “England”,

“new Delhi” to “India”,

)//Output;{Jakarta=Indoensia, London=England, New Delhi=India}

**bisa capital {“Jakarta”} bisa capital.getValue{“Jakarta”}**

1. **Muttablelist vs List**

Perbedaan dari keduanya yaitu mutablelist bisa diubah nilainya(mutable)sedangkan List tidak bias diubah nilainya(immutable).

Contoh muttablelist yang dapat diubah :

Val anylist=mutablelistOF(‘a’, “kotlin”,3,true,User())

Anylist.add(d’)//menambah item di akhir list

Anylist.add(1, “love”) // menambah item pada indeks ke-1

Anylist[3]=false//mengubah nilai item pada indeks ke-3

Anylist.removeAt(1)//menghapus item user() berdasarkan indeks atau posisi nilai didalam Array.